



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/662,720	09/15/2003	Robert T. Wickham	MS1-1552US	1329

22801 7590 08/05/2008  
LEE & HAYES PLLC  
421 W RIVERSIDE AVENUE SUITE 500  
SPOKANE, WA 99201

EXAMINER
----------

VO, TED T

ART UNIT	PAPER NUMBER
----------	--------------

2191

MAIL DATE	DELIVERY MODE
-----------	---------------

08/05/2008

PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

<b>Office Action Summary</b>	<b>Application No.</b> 10/662,720	<b>Applicant(s)</b> WICKHAM ET AL.	
	<b>Examiner</b> TED T. VO	<b>Art Unit</b> 2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 28 May 2008.
- 2a) ☐ This action is **FINAL**.                      2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-39 and 44-54 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-39 and 44-54 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- |  |   |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892)                       | 4) <input type="checkbox"/> Interview Summary (PTO-413)           |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)   | Paper No(s)/Mail Date. _____                                      |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date <u>3/28/08</u> .   | 6) <input type="checkbox"/> Other: _____                          |

**DETAILED ACTION**

1. This is in response to the communication filed on 05/28/2008.

Claims 1-39, 44-54 are pending in the application.

***Response to Amendment***

2. Regarding the amendment to claim 40-43: The amendment fails to be compliant in accordance to 37 CFR 1.121. Claims 40-43 identified the status as “Canceled”, but the text of the claims remains being shown. Applicants should be noted that all an amendment in the next reply must be pursuant to 37 CFR 1.121.

***Response to Arguments***

3. This action is in response to the amendment and argument filed on 05/28/2008.

Regarding the Applicants’ argument remarks, discussing the differences between their claims and the prior art of record based on the newly added limitations: The discussion has been considered but it is not persuasive because the discussion alleged only on the prior arts. The elements recited in the claims are claiming on top of various known ingredients, and the discussion of patentable distinction of these ingredients is absent. For example, Applicant

Art Unit: 2191

argued the reference does not teach “level of service”. However, the claims do not present a different performance on “level of service”. On the other hand, the claims tend to include common knowledge such as “grace period” used in business for allowing a thing for use in a specified period, or Failsafe timeout period, as being capable of compensating automatically and safely for a failure. These ingredients require programming techniques for doing, however, it is absent in the specification except inclusions. It should be noted that software update with these adding ingredients is merely to include the known gradients into updating. Furthermore, the lack of invention unity in the claims attempt creating burdens. Applicants’ arguments are moot in view of new ground of rejection presenting in this office action.

***Claim Rejections - 35 USC § 102***

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Art Unit: 2191

5. Claims 49-53 are rejected under 35 U.S.C. 102(a) as being anticipated by Pawlak,

“Software Update Service to Ease Patch Distribution”, April 22, 2002.

(<http://www.directionsonmicrosoft.com/sample/DOMIS/update/2002/05may/0502sustep.htm>).

As per claim 49: Pawlak further disclose, *A method implemented by a server*

*computer for performing software updates, the method comprising:*

*using a reference client computer to generate a template of approved updates* (See page:

Software Update Service Flowchart , and see page: The SUS Web Interface);

*deploying the template to a plurality of client computers* (See page: Software Update Service

Flowchart , and see page: The SUS Web Interface); *and*

*initiating software updates to the plurality of client computers according to the template* (See

page: Software Update Service Flowchart: “Start”).

As per claim 50: Pawlak further disclose, *A processor-readable medium encoded with*

*executable instructions that, when executed, direct a server computer to perform a method for*

*updating client computer performing software updates, the method comprising:*

*using a reference client computer to generate a template of approved updates* (See page:

Software Update Service Flowchart , and see page: The SUS Web Interface);

*deploying the template to a plurality of client computers* (See page: Software Update Service

Flowchart, and refer to the internet webpage of SUS Web Interface); *and*

*initiating software updates to the plurality of client computers according to the template* (See page: Software Update Service Flowchart: “Start”).

As per claim 51: Pawlak further disclose, *The processor-readable medium of claim 50, the method further comprising: incorporating the template into an XML file* (See sec. How SUS Works).

As per claim 52: Pawlak further disclose, *The processor-readable medium of claim 50, the method further comprising: deploying the template with instructions for configuring the template for SMS consumption and deployment* (See page: Software Update Service Flowchart, and refer to the internet webpage of SUS Web Interface).

As per claim 53: Pawlak further disclose, *The processor-readable medium of claim 50, the method further comprising: using the template to identify a subset of software update files from a plurality of software update files* (See page: Software Update Service Flowchart, and refer to the internet webpage of SUS Web Interface).

### ***Claim Rejections - 35 USC § 103***

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 1-12, 17-33, 44-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Pawlak, “Software Update Service to Ease Patch Distribution”, April 22, 2002.

As per claim 1: Pawlak discloses,

*A processor-readable medium encoded with executable instructions that, when executed, direct a server computer to perform a method for updating client computer, the method comprising:*

*Assigning, by a server computer, (See sec. “How SUS Works: refer to “SUS Server”) a level of service to each of a plurality of client computers (See See A.5);*

*scheduling, by the server computer, (See “Software Update Service Flowchart”, SUS server starts running scheduled synch.. in the Server-side) performance of software updates to a particular client computer from among the plurality of client computers (i.e. SUS AU client.*

Furthermore, see “Software Update Service Flowchart”, from the Server-Side, a new package for update to assigned to a client) *according to the level of service assigned to the particular client computer; and*

*initiating* (in “Software Update Service Flowchart”, i.e. “START” begun from the Server-Side processes), *by the server computer, execution of the software updates according to the schedule* (See A.1 and A.2-3).

Pawlak does not address each of the SUS AU clients is having *a level of service*. It clearly suggests the sizes and scales of service organizations or applying different group policies (See

Art Unit: 2191

sec. “Scaling SUS for Larger Organizations”). For the deficiency in mentioning service level in the reference, the difference is only in ingredients for receiving the software in the different period. It does not cause any different effect in the any particular client but updating. The update performance, even in one level or another level remains the same to the software update to SUS UA clients. It should be noted that in the MPEP, the integration or the separation of ingredients does not cause patentable difference.

Therefore, it would be obvious to an ordinary in the are at the time of filing to perform software update on the SUS AU with different types for business requirement, where the cause for doing the service on a level of service would not present patentable difference based on adding/integrating/separating ingredients.

As per claim 2: Pawlak discloses, *The processor-readable medium of claim 1, wherein the method further comprises:*

*configuring by the server computer, a postponement icon that, when displayed by the particular client computer and selected by a user, causes the execution of the software updates to be postponed within a grace period* (See A.2-3, and further see a stage in the client-side process, when Admin sees status balloon has option to defer installation: interpretable to *postponed within a grace period*), *wherein the grace period is followed by an enforcement period* (i.e. scheduled, or see page 6, “Critical Update Notification service”) *within which selection of the postponement icon is prohibited so that execution of the software updates may not be further postponed* (See A.2-3, the update is available only up to 3/14/2002).



Pawlak, does not mention “icon”, but icon is common in the art because software update/installation always includes icon such as “Next”, or “cancel” for causing installation execution. It is obvious to include because of common in the art.

As per claim 3: With the absence of mentioning *assigning the level of service comprises* but is obviousness, as being incorporated to the claims 1 and 2, Pawlak discloses, *The processor-readable medium of claim 2, wherein assigning the level of service comprises: establishing the grace period and the enforcement period; and wherein by shortening the grace period a higher level of service results due to more rapid application of the software updates* (See A.2-3).

As per claim 4: Pawlak discloses, *The processor-readable medium of claim 1, wherein the method further comprises: configuring, by the server computer, an execution icon that, when displayed by the particular client computer and selected by a user, causes the execution of the software updates to be initiated immediately* (See A.2-3).

Pawlak, does not mention “icon”, but icon is common in the art because software update/installation always includes icon such as “Next”, or “cancel” for causing installation execution. It is obvious to include because of common in the art.

As per claim 5: With the absence of mentioning *icon* but is obviousness, as being incorporated to the claim 4, Pawlak discloses, *The processor-readable medium of claim 4, wherein configuring the execution icon comprises: enabling the client computer to display a reminder about installing the software updates; and enabling the client computer to display the execution icon*. See A.2-3, left, check-boxes. And refer to the stage “Admin sees status balloon has option to defer installation”.

Art Unit: 2191

As per claim 6: With the absence of mentioning *icon* but is obviousness, as being incorporated to the claims 4-5, Pawlak discloses,

*The processor-readable medium of claim 5, wherein the reminder comprises: information on grace and enforcement periods associated with the software updates; wherein the grace period is a period during which the execution of the software updates is allowed to be postponed; wherein the grace period is configurable by a server administrator; and wherein the enforcement period is a period, configured by the server administrator to follow the grace period, during which execution of the software updates is not allowed to be postponed .* See A.2-3, and refer to the stage “Admin sees status balloon has option to defer installation”.

As per claim 7: With the absence of mentioning icon but is obviousness, as being incorporated to the claims 4-5, Pawlak discloses, *The processor-readable medium of claim 5, wherein enabling the client computer to display the execution icon; enabling an update start time to be modified by a user of the client computer; and enabling a client computer reboot time to be modified by a user of the client computer.* See A.1.

As per claim 8: Pawlak discloses, *The processor-readable medium of claim 1, wherein the method further comprises deploying annoyance reminders to the client computer urging a user of the client computer to reboot* by suggesting the Software Update Service Flowchart, a requirement for reboot, and address the limitation on SUS (See sec. Other SUS Limitations) or show a message “Installation requires a reboot”, in the page, **The SUS Web Interface**.

Pawlak does not address *deploying annoyance reminders*; but it is obvious to the ordinary in the art to include *deploying annoyance reminders*, similarly to the message “Installation requires a

Art Unit: 2191

reboot” appearing in the page. It is only for reminding to ensure a completion where reboot is a requirement for allowing update software being integrated in a computer system.

As per claim 9: Pawlak discloses, *The processor-readable medium of claim 1, wherein the method further comprises causing the client-computer to automatically perform the software updates following a grace period.* See Software Update Service Flowchart, and refer to “Admin sees status balloon has option to defer installation” and stages after it.

As per claim 10: Pawlak discloses, *The processor-readable medium of claim 1, wherein the method further comprises enabling the client computer to delay the performance until after conclusion of a user-initiated postponement within a grace period.* See Software Update Service Flowchart.

Pawlak does not address *until after conclusion*, but it is obvious to include because this is human factor where a client under SUS and has his mind to conclusion within the period the Admin sees the status balloon.

As per claim 11: Pawlak discloses, *The processor-readable medium of claim 1, wherein the scheduling comprises configuring a change time-window, wherein the change time-window defines a period of time within which client computers will not be restricted from performing the updates.* See Software Update Service Flowchart.

As per claim 12: With the absence of mentioning *assigning the level of service* but is obviousness, as being incorporated to the claims 4-5, Pawlak discloses, *The processor-readable medium of claim 11, wherein assigning the level of service comprises configuring a duration of the change time-window, wherein a longer duration implies a higher level of service and a*

*shorter duration implies a lower level of service*, inherently in mentioning Automatic Update Client”, and shown in A.1, and A.2-3.

It is obvious to include in this assigning with the configuration for conforming to such level of service.

As per claim 17: Incorporated with the rationale addressed in Claims 1, 11, Pawlak further discloses,

*The processor-readable medium of claim 11, wherein the method further comprises associating client computers into groups, wherein each group is assigned a time-change window* (See page 3, within “Automatic Update Client”, See A.1, and A.2-3).

As per claim 18: Incorporated with the rationale addressed in Claim 1, Pawlak further discloses,

*The processor-readable medium of claim 1, comprising additional instructions for: grouping a plurality of the software updates into a package; and configuring the package for differential enforcement whereby different client computers receive different software updates from the package* (Refer to “software package”, “patch” that the SUS deploys to each Scenario in the reference)

As per claim 19: Incorporated with the rationale addressed in Claims 1, 18, Pawlak further discloses, *The processor-readable medium of claim 18, wherein the method further comprises programmatically obtaining the plurality of software updates from a trusted source of update content* (See Firewall used in the reference).

Art Unit: 2191

As per claim 20: Incorporated with the rationale addressed in Claims 1, 18, Pawlak further discloses, *The processor-readable medium of claim 18, wherein the method further comprises configuring the package for SMS consumption* (See SUS/SMS used in the reference).

As per claim 21: Incorporated with the rationale addressed in Claims 1, 18, with the absence of mentioning *assigning the level of service* but is obviousness, Pawlak further discloses, *The processor-readable medium of claim 18, wherein assigning the level of service comprises providing different rules of enforcement within the package to result in different application of software updates within the package to different client computers* (See Scenarios illustrated in the reference).

As per claim 22: Incorporated with the rationale addressed in Claims 1, 18, with the absence of mentioning *assigning the level of service* but is obviousness, regarding, *The processor-readable medium of claim 18, wherein assigning the level of service comprises partitioning the package of software updates to separate trusted updates from un-trusted updates*

For the deficiency in mentioning *separate trusted updates from un-trusted updates* in the reference, the difference is only separation of ingredients for receiving the software. It does not cause any different effect in the any particular client but updating. The update performance, even in one level or another level remains the same to the software update to SUS UA clients. It should be noted that in the MPEP, the integration or the separation of ingredients does not cause patentable difference.

Therefore, it would be obvious to an ordinary in the art at the time of filing to perform separation of software on the SUS AU with different types clients for business requirement,

where the cause for doing the service on dividing would not present patentable difference based on adding/integrating/separating ingredients.

As per claim 23: Incorporated with the rationale addressed in Claims 1, 18, 22 with the absence of mentioning *assigning the level of service* but is obviousness, Pawlak further discloses, *The processor-readable medium of claim 22, wherein assigning the level of service further comprises merging the un-trusted software updates with the trusted software updates based on performance of the un-trusted updates in a test environment* (Refer to Patch distribution, and see Scenarios illustrated in the reference).

As per claim 24: Incorporated with the rationale addressed in Claims 1, 18, and 22, regarding, *The processor-readable medium of claim 22, wherein the partitioning is expressed in XML configured to inform different clients of updates suitable for their consumption* (Note: XML is common and in public uses. Pawlak shows it).

As per claim 25: Incorporated with the rationale addressed in Claim 1, with the absence of mentioning *assigning the level of service* but is obviousness, Pawlak further discloses, *The processor-readable medium of claim 1, wherein assigning the level of service comprises: incorporating an authorization list of approved updates into a template based on a standard image* (See A.1. A-2-3).

As per claim 26: Incorporated with the rationale addressed in Claims 1, 25, Pawlak further discloses, *The processor-readable medium of claim 25, wherein the template is written into an XML document* (Note: The claims is only to conform to the file format of HTML/ XML).

Art Unit: 2191

As per claim 27: Incorporated with the rationale addressed in Claims 1, 25-26, Pawlak further discloses, *The processor-readable medium of claim 26, wherein the XML document is consumed and deployed as a mirror of a desired state for software updates* (See A.1).

As per claim 28: Incorporated with the rationale addressed in Claims 1, 25-27, Pawlak further discloses, *The processor-readable medium of claim 27, wherein the XML document is consumed and deployed by SMS* (See A.1).

As per claim 29: See rationale addressed in the claim 1.

As per claim 30: See rationale addressed in the claims 1-2.

As per claim 31: Incorporated with the rationale addressed in Claim 30, Pawlak further discloses, *The processor-readable medium of claim 30, wherein the method further comprises: providing a client computer user interface at repeated intervals to facilitate the reboot of the client computer, where the software updates have been installed and no reboot has been performed.* See A.1.

As per claim 32: Incorporated with the rationale addressed in Claim 30, Pawlak further discloses,

*The processor-readable medium of claim 30, wherein the method further comprises: setting the grace periods and the enforcement periods to control a level of service provided to the client computer.* See A.1.

As per claim 33: Incorporated with the rationale addressed in Claim 30, Pawlak further discloses, *The processor-readable medium of claim 30, wherein the method further*

*comprises: periodically displaying information about software updates that have not yet been performed.* See A.1, A2-3.

As per claim 44: Pawlak discloses,

*A method executed by a server system for performing client computer software updates, the method comprising:*

*forming a package with a plurality of software updates;*

*partitioning the package to divide trusted updates from un-trusted updates;*

*distributing the package to a plurality of client computers, such that appropriate software updates are installed on each of the plurality of clients, wherein the un-trusted software updates are installed only on client computers configured by the server to install un-trusted software.*

See section The Need for Automated Patch Distribution, and A1-3, A5-6, and refer to the rationale addressed in the claim 1.

Pawlak does not address each of the SUS AU clients is *divide trusted updates from un-trusted updates*. It clearly suggests the type of service organizations or applying different group policies (See sec. “Scaling SUS for Larger Organizations”). For the deficiency in mentioning *divide trusted updates from un-trusted updates* in the reference, the difference is only separation of ingredients for receiving the software. It does not cause any different effect in the any particular client but updating. The update performance, even in one level or another level remains the same



to the software update to SUS UA clients. It should be noted that in the MPEP, the integration or the separation of ingredients does not cause patentable difference.

Therefore, it would be obvious to an ordinary in the art at the time of filing to perform software update on the SUS AU with different types clients for business requirement, where the cause for doing the service on dividing would not present patentable difference based on adding/integrating/separating ingredients.

As per claim 45: See rationale addressed in Claim 44.

As per claim 46: Pawlak further disclose,

***merging un-trusted software updates together with the trusted software updates in response to performance of the un-trusted software updates on clients having un-trusted software updates installed.*** Refer patching distribution.

As per claim 47: Pawlak further disclose,

***The processor-readable medium of claim 45, the method further comprising: expressing the partitioning of the package with XML.***

See related rationale addressed in claim 24.

As per claim 48: Pawlak further disclose, The processor-readable medium of claim 45, the method further comprising: embedding within the package, instructions that when executed by the client computer, facilitate the expressing to client computers which software updates are suitable for their consumption. (Refer to patch distribution with package, and see the page Software Update Service Flowchart).

8. Claims 13-16, 34-39, 54 are rejected under 35 U.S.C. 103 (a) as being unpatentable over Pawlak, “Software Update Service to Ease Patch Distribution”, April 22, 2002, in view of IBM, “RS/6000 ATM Cookbook”, Redbook.ibm.com, 2000.

As per claim 13: Regarding, *The processor-readable medium of claim 11, wherein the scheduling further comprises: defining failsafe timeout periods for each of the software updates; and adjusting the failsafe timeout periods according to individual client computer performance, wherein longer failsafe timeout periods are assigned where the individual client computer performance is slower.*

Pawlak discloses update scheduling, but does not address “*failsafe timeout period*”

However, “failsafe timeout period” is very common in installation/updating for terminating a process in which the timing exceeds predetermined maximum if the process requires time limit.

The IBM reference defines failsafe timeout period as a maximum time in second that allows a client to recover from network outage.

Therefore, it would be obvious to an ordinary of the art at the time of the application filing to include the “failsafe timeout period” in the update for stopping wasting unnecessary time when it knows that the update could take timing that less than a predetermined maximum. This type of act is done common in installing/updating. For example, IBM has shown a grace period that has been set in an installation of ATM software (See IBM, p. 32, and p. 151).

As per claims 14-16:

Claim 14, regarding, *The processor-readable medium of claim 11, wherein the method further comprises: applying updates during the change time-window; and monitoring a failsafe timeout for each update applied.*

Pawlak discloses update scheduling, but does not address “failsafe timeout period”;

Claim 15: Regarding, *The processor-readable medium of claim 11, wherein the method further comprises identifying updates for which there was insufficient time within the change time-window for installation within a second change time-window.*

Pawlak discloses identify update, but does not address “failsafe timeout period”;

Claim 16: Regarding, *The processor-readable medium of claim 11, wherein the method further comprises, when time remaining within the change time-window is less than a failsafe timeout for any remaining software updates, suspending application of the remaining software updates.*

Pawlak discloses suspending (See page Software Update Service Flowchart: Admin see...has option to defer installation), but does not address “failsafe timeout period”

However, “failsafe timeout period” is very common in installation/updating for terminating a process in which the timing exceeds predetermined maximum if the process requires time limit.

The IBM reference defines failsafe timeout period as a maximum time in second that allows a client to recover from network outage.

Therefore, it would be obvious to an ordinary of the art at the time of the application filing to include the “failsafe timeout period” in the update for stopping wasting unnecessary time when it knows that the update could take timing that less than a predetermined maximum. This type of act is done common in installing/updating. For example, IBM has shown a grace period that has been set in an installation of ATM software (See IBM, p. 32, and p. 151).

As per claim 34: Pawlak discloses

*A method executed by a server computer for performing software updates on client computers, the method comprising: associating client computers into groups (See sec. Scaling SUS for Larger Organizations, see page Software Update Service Flowchart); establishing a change time-window for each of the groups (Refer to Client-side processes); and initiating, by the server computer, software updates to the client computers within the change time-window established for each group of client computers (Refer to Client-side processes); monitoring by the server computer a failsafe timeout for each update.*

Pawlak’s update does not address monitoring by the server computer a failsafe timeout

The IBM reference defines failsafe timeout period as a maximum time in second that allows a client to recover from network outage.

Therefore, it would be obvious to an ordinary of the art at the time of the application filing to include the “failsafe timeout period” in the update for stopping wasting unnecessary time when it knows that the update could take timing that less than a predetermined maximum.

Art Unit: 2191

This type of act is done common in installing/updating. For example, IBM has shown a grace period that has been set in an installation of ATM software (See IBM, p. 32, and p. 151).

As per claim 35: See rationale addressed in Claim 34.

As per claim 36: Pawlak discloses *The processor-readable medium of claim 35, the method further comprising: installing each software update* (See the Software Update Service Flowchart);

but does not disclose *and setting the failsafe timeout with reference to the anticipated duration of installation*; see definition of failsafe, in IBM.

As per claim 37: Pawlak discloses *The processor-readable medium of claim 35, the method further comprising: determining, by the server computer, if the failsafe timeout for each software update is greater than time remaining within the change time-window, and if so, suspending installation of the software update.* See page Software Update Service Flowchart: Admin see...has option to defer installation, but does not address “*failsafe timeout period*”

IBM discloses *if the failsafe timeout*. See definition of failsafe, in IBM.

As per claim 38: Pawlak further discloses *The processor-readable medium of claim 35, the method further comprising: identifying, by the server computer, software updates for installation in a second change time-window, which were not installed in the change time-window* (See the mentioning the option to defer installation).

As per claim 39: Pawlak further discloses *A method executed by a server computer for performing software updates, the method comprising:*

*grouping a plurality of software updates into a package;*

*configuring the package for differential enforcement, wherein different client computers are assigned by the server different periods of time within which a software update will be initiated; and configuring the package for SMS consumption.*

(See the page Software Update Service Flowchart, in the Server-Side Processes, and sec. How SUS Works);

As per claim 54: Regarding limitations: *A processor-readable medium comprising processor-executable instructions for performing software updates, the processor-executable instructions comprising instructions for:*

*configuring the package with content from a trusted website;*

*grouping a plurality of software updates into a package;*

*configuring the package for SMS consumption;*

*partitioning the package to divide trusted updates from un-trusted updates;*

*distributing the package by utilizing SNS to a plurality of client-computers;*

*associating client-computers into groups;*

*establishing a change time-window for each of the groups;*

*expressing to client-computers which software updates are suitable for consumption;*

*installing updates on each of the plurality of clients within the change time-window established for the group the client is a member of;*

*installing the un-trusted software updates are installed only on client-computers configured to install un-trusted software updates;*

Art Unit: 2191

*setting a failsafe timeout with reference to the anticipated duration of installation;*

*monitoring a failsafe timeout for each update;*

*determining if the failsafe timeout for each software update is greater than*

*time remaining within the change time-window, and if so, for suspending*

*installation of the software update.*

With regard to the recitation above, see the rationale and the obviousness as addressed in the rejection of claims 1-12, 18-28, and addressed further in Claim 34 above.

With regard to the limitation *failsafe timeout*, see addressing obviousness in combination in Claim 34 above.

### ***Conclusion***

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ted T. Vo whose telephone number is (571) 272-3706. The examiner can normally be reached on 8:00AM to 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Y. Zhen can be reached on (571) 272-3708.

The facsimile number for the organization where this application or proceeding is assigned is the Central Facsimile number 571-273-8300.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100. Information regarding the status of

Art Unit: 2191

an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

TTV  
July 25, 2008

/Ted T. Vo/  
Primary Examiner, Art Unit 2191